REMARKS

Applicant thanks the Examiner for his careful consideration of this application.

Claims

It is noted that the claims as amended by this present Amendment and Response to Office Action are amended from the set of claims filed prior to the filing of the Amendment After Final Under Rule 116, since the Amendment After Final Under Rule 116 was not entered. Accordingly, the claims as amended herein are amended from the set of amended claims which were filed in the Amendment mailed May 23, 2005 (certificate of mailing).

Independent claim 1 has been amended for clarification by reciting, "a client/server application comprising a server application component and a client component", "the server application component", "determining the appropriate portion of the plurality of portions to load and execute on the server computer to provide the selected function to the client computer", and "whereby the portions are incrementally loaded and executed on the server computer".

Independent claims 3, 5 and 7 have been amended to add changes corresponding to those of claim 1. The amendment to the claims is fully supported by the application as originally filed, for example, on page 5, line 26-page 6, line 2, page 6, lines 32-34 and in Figure 3.

The description on page 5, line 26-page 6, line 2 states:

> -- ...one portion of the plurality of portions 20 is a compact portion 28 initially executed upon receipt of a first application function request from the client component 18. Execution of the application occurs in states, with a compact portion 28 of the application begin initially invoked to provide a streamlined subset of functions applicable to commands most commonly requested to provide a fast executing initial portion of the application. Should the function requested lie outside of the function set of the compact portion 28, an applicable additional executable portion is loaded and executed, combining its function(s) with the compact portion's 28 functions, with minimal overlap, to provide enhanced functionality.--[Emphasis added]

6

The description on page 6, lines 32-34 states:

> -- the invention enables a large decrease in the number of lines of script at initial runtime when compared with a full version of <u>an application</u>--[Emphasis added]

In this application, "application" runs in the server computer, and not in the client. As disclosed on the description on page 5, line 30, and elsewhere, the "invoking of the portion" would be understood by those persons skilled in the art as including the steps of loading the portion into execution memory from some form of backing store (such as a disk drive) and then causing its execution, including initialising.

Thus it is respectfully submitted that no new matter has been introduced by way of the amendment to claims 1, 3, 5 and 7.

Dependent claims 2, 4, 6 and 8 have been amended to delete the term "streamlined" as being redundant in view of the specification, for example, on page 5, lines 29-32, which states: " ... a compact portion 28 of the application being initially invoked to provide a streamlined subject of functions ...". No new matter has been introduced by way of the amendment to claims 2, 4, 6 and 8.

New dependent claim 9 has been added. As discussed above, the "invoking of the portion" would be understood by those persons skilled in the art as including the steps of loading the portion into execution memory from some form of backing store (such as a disk drive) and then causing its execution, including initialising. Applicant believes that no new matter has been introduced by way of the amendment.

The Examiner rejected claims 1-8 under 35 U.S.C. 102(e) as being anticipated by Kevner (U.S. Patent No. 5,956,509).

Applicant respectfully requests reconsideration and withdrawal of the rejections, because Applicant believes that claims 1-9 are patentable in view of the cited reference for reasons set out below.
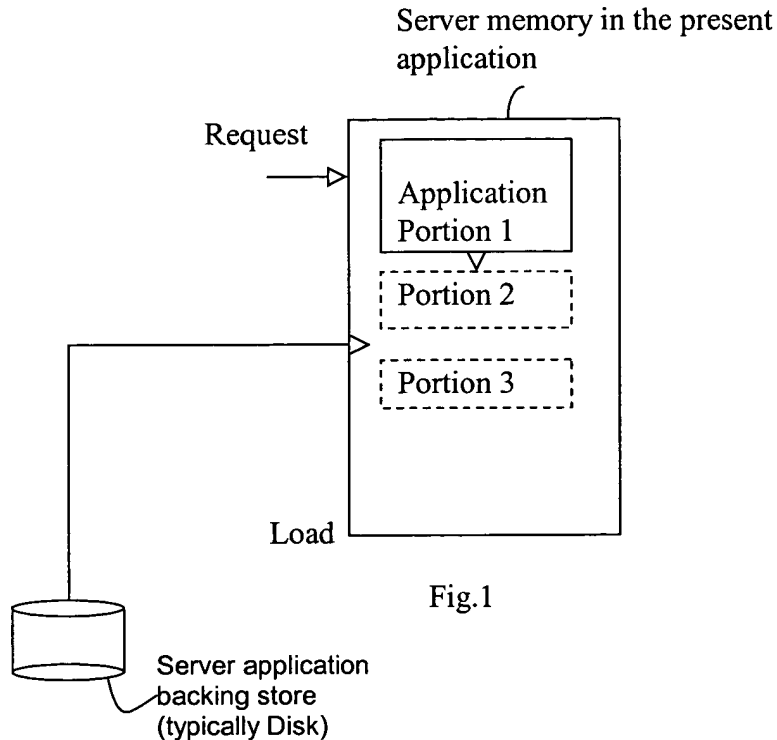
The present application relates to the dividing up of a server application component into

portions so that the amount of code required to be <u>loaded initially into the server computer</u> is minimized, <u>but provides the minimum functionality required in the server computer</u>. The loading of further portions of code to add functions is determined by what functions are required by the client - these map to particular portions of code in the server. As a result, the functions requested by the client computer are incrementally provided to the client computer in response to one or more request commands from the client computer.

On page 5 of the Final Office Action, the Examiner stated that Kevner teaches incrementally executing the service applications 200b on the server as requested by the client (col. 8, lines 14 -21 of Kevner).

Applicant respectfully submits that col. 8, lines 14-21 of Kevner merely teaches that a user can request various separate applications to be executed, or maybe which function within an application.

By contrast, according to the present invention, the portions of the server application component (which is an application divided up into portions) are <u>incrementally loaded and executed on the server computer as required</u>. For Examiner's review, Applicant attaches Fig. 1 below, which further illustrates the concept of "loading and "executing" disclosed in the present application. The initial request for a function results in the Portion 1 of the application being loaded into memory and executed. When any function not able to be performed by this first portion is requested, the appropriate portion is loaded and executed. This loading of the application in portions (incremental loading) is the result of careful design of a system that ensures that server resources (including memory) are used optimally, and that users (of the client machines) perceive a faster loading of such applications.

Server memory in the present
application

Request

Application
Portion 1

Portion 2

Portion 3

Load

Fig.1

Server application
backing store
(typically Disk)

Kevner neither discloses nor suggests incrementally loading and executing the portions of the
server application component on the server computer, as illustrated in the above Fig.1.

The present invention addresses the problem of execution times (or rather response times as
perceived by a user) for a client/server application. It is well known that as application software
has become more sophisticated and included more and more functions, the time for initializing
and preparing it for operation, including loading into memory when necessary, (all of which are
included in the execution time) has increased compared to earlier, simpler applications. Even
on today's large servers providing access to corporate databases using business reporting
software, the processes of loading and initialization during execution results in extended delays.
To reduce the execution times for a client/server application, a system and method in
accordance with the present invention incrementally loads and executes, on the serve computer,
the portions of the application, as discussed above and explained, for example, on page 5, lines
29-32 of the present application. When read in conjunction with the background, which on
page 2, lines 4-6, states "What is needed is a method of providing a reduction in execution times
for client/server applications without the requirement for separate, resource intensive
components...", it is clear that the feature of "incrementally loading and executing" is to

9

provide improvements at the server, not at the client alone.

By contrast, Kevner, among other things, provides for the structured <u>loading of information</u> <u>(data and code) onto the client</u>. Kevner teaches that applications on the server can allow the client to continue while waiting for a response from the server. Kevner embraces the use of client applications, something specifically considered not suitable for the environment of the present invention: on page 1, line 30-page 2, line 6 which deprecatingly refers to "… heavily coded client…" etc.

On page 5 of the Final Office Action, the Examiner stated that Kevner does teach parameters associated with remote requests from client to server (col. 13, lines 13 -21 of Kevner).

Applicant respectfully submits that according to the present invention, a parameter is used to determine which portion of the application in the server application component is to be <u>loaded</u> <u>and executed on the server computer</u>. Kevner on col. 13, lines 13-21 states that the parameters 424 contain data related to the remote request or indicate that the parameters are executed to be returned from a service application 200b. However, there is no suggestion or teaching on col. 13, lines 13-21 of Kevner that the parameters 424 is used to incrementally loading and executing, on the server computer, the portions on the server application component.

Applicant respectfully responds to each of the Examiner's points on pages 2-4 of the Final Office Action in turn:

The Examiner stated that Kevner discloses, on col. 4, lines 30-35, "a system for incrementally executing a client/server application" in claim 1. Applicant respectfully submits that col. 4, lines 30-35 of Kevner states the transfer of datablocks between the client and server, and neither suggests nor teaches incremental execution of an application.

The Examiner stated that Kevner discloses, on col. 7, lines 1-5, "a server component comprising a plurality of portions, …" in claim 1. Applicant respectfully submits that col. 7, lines 1-5 of Kevner merely states "each client-server application includes a client portion and <u>a</u> server portion". By contrast, as discussed above, it is clear that according to claim 1, more than one portion is envisaged in the server application component for a server computer.

The Examiner stated that Kevner discloses, on col. 28, lines 12-17, "associated code ..." in claim 1. Applicant respectfully submits that col. 28, lines 12-17 of Kevner discloses a command which relates to the selection of a client CHAT application. By contrast, "associated code ..." in claim 1 is for selection from a plurality of portions in the server application components.

The Examiner stated Kevner discloses, on col. 28, lines 1-5 and col. 13, lines 13-21, "an associated parameter ..." in claim 1. Applicant respectfully submits that col. 28, lines 1-5 of Kevner discloses a function in the client, i.e., the RequestDynamicParam routine. Applicant respectfully submits that col. 13, lines 13-21 of Kevner discloses parameters relating to the remote request, however, it does not disclose how they might be used. By contrast, "an associated parameter ..." in claim 1 refers to a parameter used to determine which portion of the application is loaded and executed on the server computer.

The Examiner stated Kevner discloses, on col. 3, lines 30-37, col. 4, lines 30-37 and col. 8, lines 4-21, "the portions are incrementally executed ..." in claim 1. Applicant respectfully submits that Kevner in col. 3, lines 30-37, col. 4, lines 30-37 merely relates to the transfer of datablocks between the client and server. Applicant respectfully submits that col. 8, lines 4-21 of Kevner relates to the execution of the application in the server, and neither suggests nor teaches the feature of incremental loading and execution of such an application.

With respect to the objection to claim 2, the Examiner referred to col. 9, lines 42-55 of Kevner. It is noted that the present application deals with the initial execution of portions on the server computer. Col. 9, lines 42-55 of Kevner relates to the distribution of loads between servers, and makes a general statement that users can and do access multiple services throughout a session.

With respect to the objection to claim 3, the Examiner referred to col. 16, lines 45-60 and col. 45, lines 5-20 of Kevner. It is noted that the present invention is concerned with the portions of a server application component in the server. Col. 16, lines 45-60 of Kevner relates to sending data to the server-multiple packets of 400 or fewer bytes, and is concerned with memory allocation within the client. Col. 45, lines 5-20 of Kevner merely states that control of these packets uses bits embedded in the packets, and is not related to incremental loading and

11

executing the portions of the server application component on the server computer.

With respect to the objection to claim 4, the Examiner referred to col. 8, lines 35-50 of Kevner. Applicant respectfully submits that col. 8, lines 35-50 of Kevner is concerned with aspects of client operation, and not server operation

The above arguments of claims 1, 2, 3 and 4 are applied to overcome the objections to claims 5, 6, 7, and 8.

It is respectfully submitted that claims 1-8 are patentable in view of the cited reference. Applicant respectfully requests the Examiner to withdraw the rejections.

In view of the above amendments and remarks and having dealt with all the objections raised by the Examiner, reconsideration and allowance of the application is courteously requested.

If any fees are required by this Amendment and Response to Office Action or the accompanying Request for Continued Examination (RCE) Transmittal which are not covered by an enclosed check, please charge any such fees to our Deposit Account No. 16-0820, Order No. 33263US1.

Respectfully Submitted,

PEARNE & GORDON LLP

By _____
Steven J. Solomon, Reg. No. 48719

1801 East 9th Street
Suite 1200
Cleveland, Ohio 44114-3108
(216) 579-1700

Date: April 7, 2006

12